

**Concordia University
Department of Computer Science
and Software Engineering**

**Advanced Programming Practices
SOEN 6441 --- Winter 2016**

Project Description

Deadline:	Intermediate delivery 1: February 18 th , 2016 Intermediate delivery 2: March 17 th , 2016 Final delivery: April 7 th , 2016
Evaluation:	Intermediate delivery 1: 15% of final mark Intermediate delivery 2: 15% of final mark Final delivery: 20% of final mark
Late submission:	not accepted
Teams:	the project is a team assignment

General description

The project is to be undertaken teams of exactly 5 members and consists of the building of a challengingly large Java program. The completion of the project is divided into three separate components: (1) the *First and Second Intermediate Project Delivery* are intermediate operational build of the software, effectively demonstrating the full implementation of some important software features; (2) the *Final Project Delivery* is the demonstration of the finalized version of your software. During the final project delivery, you also have to demonstrate that your code includes many of the Java features presented in the lectures, and that you effectively use the tools presented in the lectures in your project. All project deliveries are to be undertaken in the laboratory where the team presents the implemented features to the instructor following a pre-circulated grading sheet. The individual assignments will also be related to the project, but graded individually and separately from the project.

It is important to realize that the project description is purposely incomplete, and that it is one of your duties in this project to: 1) elicit and formulate all the missing details before you start the implementation, 2) limit the scope of the project according to the time that is available, 3) determine what design decisions will be made, as well as 4) what tools will be used for the implementation. These activities require some investigations and discussions that are important aspects of software development and this project.

Problem statement

The specific project for this semester consists in building a simple "Tower Defense" computer game. Examples of such a game includes "Flash Element TD" and "Vector TD", which can be played online for free. A Tower Defense game consists of a grid map onto which there is a path connecting an entry point and an exit point. At each game "wave", critters are coming onto the map from the entry point and follow the path to the exit point. The player is to place "towers" outside of the path that are shooting at the critters as they follow the path. The objective of the game is to destroy the critters before they reach the exit point. The strength of the critters increases at every wave, forcing the player to increase the number and strength of the towers placed on the map. Destroying a critter is worth a certain number of points, which can be used to buy towers. Different kinds of towers and critters have different characteristics.

Expected parts/features

Map: The game map consists of a grid onto which there is a path connecting the entry point to the exit point. Critters are allowed to move on the path, and towers are allowed to be placed only outside of the path, which is called the scenery. A map editor allows the user to create specific maps by specifying the size of the map (e.g. 10x10, 5x30, etc), specifying what cells are the entry point and the exit point, and which cells are part of the path, or part of the scenery. A valid map is a map that has an entry point and exit point that are connected by path cells. Once a valid map has been created, it can be played using the game (see below). During play, any given cell may hold only one game element (tower or critter). Path cells may contain only critters, and scenery cells may contain only towers.

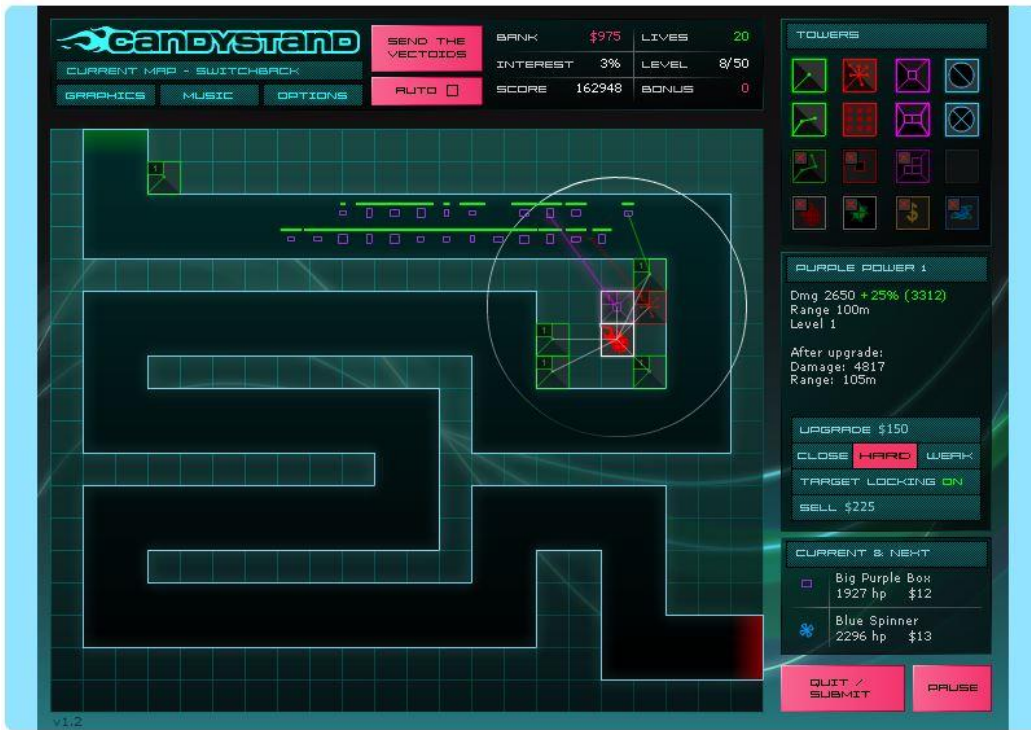
Game: The game starts by allowing the player to select a map to play on. Once the map is selected, the player is given a certain amount of initial currency points to buy a few towers of his choice and place them on the map. The game is “wave-based”. When the player signifies that he is ready for the next wave, a finite group of critters appear in succession from the entry point and start heading to the exit point along the path. As the critters get in the range of the towers, the towers shoot at them as the critters move along the path, inflicting damage. Critters that receive more damage than they have hit points are removed from the game. The game ends when a certain number of critters are able to reach the end point before they are destroyed by the towers. If all critters in a wave are destroyed, this wave is over and the player prepares for the next wave.

Towers: The towers can be bought by the player at any time during play. Each kind of tower has a different cost, with more potent towers costing more points to buy. The player is able to inspect the characteristics of every kind of tower, but is only able to buy those that he can afford. Once a tower is bought, the player is allowed to place it on the map, but only on the scenery cells, i.e. outside of the path. Once the tower is placed on the map, it becomes effective. Every tower has a range, outside of which it cannot shoot. It has a power, which is the amount of damage that it inflicts on a critter every time it hits. It also has a rate of fire, which determines the amount of time that must elapse before it can shoot again. More elaborated towers may have special effects such as area of effect damage or critter stunning which slows critters after a hit. Towers that are on the map can be inspected to show their characteristics and upgraded (for a certain amount of currency points) to increase their capacity, or sold and removed from the map in exchange for some currency points.

Critters: As soon as the player signifies that he is ready for the next wave, this wave’s group of critters is created, appearing in succession at the starting point, after which they start moving along the path towards the ending point. As the critters move, they may come within the range of some of the towers and be fired upon, resulting in damage that is determined by the tower’s power. A critter that sustains more damage than its maximum hit points is destroyed and removed from the map. Each critter destroyed earns a certain amount of currency points to the player. Once a certain number of critters have reached the ending point without being destroyed, the game is over. If all critters in a wave are destroyed, this wave is over and the player prepares for the next wave. The strength of the critters increases at every wave, forcing the player to place more towers. The player should be able to inspect the characteristics of the critters in play, as well as the characteristics of the critters to come in the following waves.

Example Games

Candystand's "Vector TD": <http://www.ineedtoplay.com/full-219-Vector-TD-2.html>



Kongregate's "Flash Element TD 2": <http://www.kongregate.com/games/casualcollective/flash-element-td-2>

